

Magic

Input file: **standard input**
Output file: **standard output**
Time limit: **3 seconds**
Memory limit: **16 megabytes**

Warning: Unusual memory limit!

You are given a sequence $a_0, \dots, a_{2n}$. Initially, all numbers are zero.

There are n operations. The i -th operation is represented by two integers l_i, r_i ($1 \leq l_i < r_i \leq 2n, 1 \leq i \leq n$), which assigns i to $a_{l_i}, \dots, a_{r_i-1}$. It is guaranteed that all the $2n$ integers, $l_1, l_2, \dots, l_n, r_1, r_2, \dots, r_n$, are distinct.

You need to perform each operation exactly once, in arbitrary order.

You want to maximize the number of i ($0 \leq i < 2n$) such that $a_i \neq a_{i+1}$ after all n operations. Output the maximum number.

Input

The first line contains an integer n ($1 \leq n \leq 5 \times 10^3$).

The i -th line of the next n lines contains a pair of integers l_i, r_i ($1 \leq l_i < r_i \leq 2n$). It is guaranteed that all the $2n$ integers, $l_1, l_2, \dots, l_n, r_1, r_2, \dots, r_n$, are distinct.

Output

Output one integer representing the answer in one line.

Example

standard input	standard output
5 2 3 6 7 1 9 5 10 4 8	9